

OIC#2 - Programmeren in C#2

Moduleboek

2022-2023
Versie 2.8 / 03-03-2023

Programmeren in C#2

Versie 2.8
03-03-2023

Studiejaar 2022-2023



Module coördinator

Winnie van Schilt
winnie.van.schilt@nhlstenden.com

Docenten

Rob Loves
rob.loves@nhlstenden.com

Voorwoord

In deze module ga je opnieuw aan de slag met C#. Het is de bedoeling om een applicatie projectmatig in C# te ontwikkelen. We gaan in deze module gebruik maken van de laatste versie van Visual Studio.

Het is de bedoeling om de applicatie projectmatig in een team te ontwikkelen. Het doel van de applicatie is dat hij beschikt over meerdere schermen. De groep gaat zelf onderzoeken welke kennis nodig is om deze applicatie te ontwikkelen in C#. Tijdens de module zijn er een aantal momenten dat er wordt gekeken naar de voortgang van het team.

Verder is het van belang dat het team de bouw van de applicatie projectmatig en professioneel aanpakt. Hier zijn een paar verplichtingen aan verbonden waaronder een klassendiagram. Tijdens de oplevering van de App wordt hier ook weer aandacht aan geschonken.

De studielast voor deze module bedraagt 84 uur (3 EC).

Uit de verschillende toegepaste methodieken op de NHL Stenden hogeschool is door onderzoek in het verleden gebleken dat C# een geschikte methode is voor het leren programmeren voor een beginnend programmeur. Door evaluaties van studenten is het programma zodanig aangepast dat er meer tijd beschikbaar is voor het onderdeel programmeren in C#2.

Jan Doornbos en Rob Smit
Emmen, 23 maart 2022

Inhoud

1	Introductie	1
1.1	Competenties	2
1.2	Module thema	3
1.3	Module doelstellingen	3
1.4	Voorkennis	3
1.5	Conventies	3
1.6	Versiebeheer	3
2	Toetsing	5
2.1	Actieve participatie	5
2.2	Module herkansing	5
3	Programma	7
3.1	Collegevormen	7
3.2	Programmaoverzicht	8
3.3	Weekprogramma's	9
4	Structuur & Organisatie	12
5	Literatuur / software	13
5.1	Verplichte leerstof	13
5.2	Naslagwerk	13
5.3	Programmatuur	13
6	Module evaluatie	14

7	Bijlagen	15
7.1	Projectopdracht	15
7.2	Scoring Rubrics	17
7.3	Scoring Rubrics Presentatie	18
7.4	Scoring Rubrics Eindcijfer	20

1 Introductie

Object georiënteerd programmeren is één van de basiselementen voor de informaticus van dit moment. Het object georiënteerd programmeren zal in dit geval aangeboden worden in één van de meest gebruikte talen: C#.

Kenmerkende beroepssituatie

Willeke is na haar studie aan de slag gegaan als junior software engineer bij de firma Kooijman software. Willeke heeft een HBO Informatica opleiding gevolgd met als afstudeerrichting Software Engineering. Tijdens de opleiding heeft Willeke gewerkt met verschillende programmeertalen waaronder C#, Java en PHP. Tijdens haar opleiding bleek dat Willeke bijzonder geschikt was als projectleider met goede communicatievaardigheden. Tijdens de opleiding stond het werken in teams aan projecten centraal.

De werkgever van Willeke is gespecialiseerd in het op maat ontwikkelen van automatiseringsoplossingen voor de bouw. Het in eigen huis ontwikkelde bedrijfsinformatiesysteem is erop gericht het gehele bouwproces te ondersteunen zowel op proces als op administratief niveau. Het afgelopen jaar heeft Willeke volledig meegedraaid in het migratie project om de huidige software te migreren naar C#. Willeke haar belangrijkste taak in het project was de vernieuwing van de webinterface voor het bedrijfsinformatiesysteem.

Willeke kan zelfstandig en in teams werken. Ze kan communiceren met de verschillende partijen. Ze heeft een brede technische kennis en een goed inlevingsvermogen met betrekking tot wat de klant wil. Zij kan zich op verschillende gebieden snel inwerken en beheerst verschillende ontwerpmethoden, programmeertalen en ontwikkeltools om software te ontwikkelen.

Willeke kan zich op middellange termijn ontwikkelen tot senior software engineer en op lange termijn tot projectleider.

1.1 Competenties

Binnen deze module wordt als beginnend beroepsprofessional gewerkt aan een negental competenties die betrekking hebben op het beheren, analyseren, ontwerpen en realiseren van applicaties.

	Beheren	Analyseren	Adviseren	Ontwerpen	Realiseren
Gebruikers-Interactie	<ul style="list-style-type: none"> > Het toepassen van versiebeheer en het inrichten en configureren van een samenwerkings-omgeving bij de realisatie van ict- en/of digitale mediaproducten, rekening houdend met onderhoudbaarheid en daarvoor beschikbare middelen. (niveau 2)	<ul style="list-style-type: none"> > Het designrepertoire inventariseren en de merk- of productidentiteit beschrijven voor een actueel ict- en/of digitaal mediaproduct. > Doelgroepen en doelstellingen onderscheiden en deze koppelen aan gebruikersgedrag en interactie. (niveau 1)		<ul style="list-style-type: none"> > Ontwerpen van statische en beperkt dynamische ict- en/of digitale mediaproducten met toepassing van gegeven technieken, zoals scenario's, storyboards en wire frames en passende usability-testen. (niveau 1)	<ul style="list-style-type: none"> > Het realiseren en testen van statische ict- en/of digitale mediaproducten met inzet van gangbare tools, keuze voor geschikte (media-) formaten en beperkt toepassen van stijlmiddelen. (niveau 1)
Bedrijfsprocessen					
Infrastructuur					
Software	<ul style="list-style-type: none"> > Inrichten en gebruikmaken van beheersysteem ter ondersteuning van softwareontwikkeling in teamverband. (niveau 1)	<ul style="list-style-type: none"> > Een requirement-analyse uitvoeren voor een softwaresysteem met verschillende belanghebbenden, rekening houdend met de kwaliteits-eigenschappen. > Een analyse uitvoeren om functionaliteit, ontwerp, interfaces e.d. van een bestaand systeem of bestaande component te formuleren en te valideren. > Een acceptatietest opstellen aan de hand van kwaliteits-eigenschappen. (niveau 2)	<ul style="list-style-type: none"> > Aanbevelingen doen over specifieke requirements van een software-systeem op grond van onderzoek naar bestaande, vergelijkbare systemen. (niveau 1)	<ul style="list-style-type: none"> > Opstellen van een ontwerp voor een softwaresysteem, rekening houdend met het gebruik van bestaande componenten en libraries; gebruik maken van ontwerp-kwaliteitscriteria. > Vaststellen van de kwaliteit van het ontwerp, bijvoorbeeld door toetsing of prototyping, rekening houdend met de geformuleerde kwaliteitseigenschappen. > Testontwerpen opstellen volgens een gegeven teststrategie. (niveau 2)	<ul style="list-style-type: none"> > Bouwen en beschikbaar stellen van een softwaresysteem dat bestaat uit meerdere subsystemen, hierbij gebruik makend van bestaande componenten. (niveau 2)
Hardware interfacing					

1.2 Module thema

In deze module gaat een team aan studenten een concept voor een applicatie vertalen naar een werkende applicatie.

1.3 Module doelstellingen

Na het succesvol afronden van deze module kan de student:

- > Een gegeven werkberoepssituatie analyseren;
- > Een traject opzetten voor het ontwikkelen van een C# applicatie;
- > Een object georiënteerde C# applicatie ontwikkelen aan de hand van voorgeschreven criteria;
- > Vooraf bepalen welke (mogelijke) frameworks nodig zijn voor de te ontwikkelen applicatie.

1.4 Voorkennis

De vereiste voorkennis voor het volgen van de module is opgenomen in de meest recente Onderwijsbeschrijvingen van het OER.

1.5 Conventies

De te hanteren conventies met betrekking tot naamgeving en indeling van de programmacode kun je op Blackboard vinden binnen ICT-algemeen.

1.6 Versiebeheer

Versie	Datum	Auteur	Omschrijving
1.0		J. Pijpker	Opzet moduleboek.
2.0	11-04-2016	W. van Schilt	Opname versiebeheer; Update persoonsgegevens en datumrevisie; Competenties toegevoegd en aangepast aan hbo-i competentiematrix 2014; Div. wijzigingen doorgevoerd aan o.a. programma.
2.1	07-11-2016	W. van Schilt	Niveauijziging competentie; Aangepast aan nieuwe studiejaar.
2.2	12-01-2017	W. van Schilt	Competentiematrix geüpdate.
2.3	26-03-2018	R. Smit	Datumrevisie en minimale edits.
2.4	28-05-2018	R. Smit	Revisie voorwoord; Update tabel 4.1.
2.5	14-04-2020	R. Smit	UWP verplichting vervangen door WPF; Minieme aanpassingen aan rubric;

			Datumrevisie.
2.6	09-04-2021	J. Doornbos	Nieuwe layout; Aangepast aan nieuwe studiejaar.
2.7	23-03-2022	R. Smit	Aangepast aan nieuwe studiejaar. Tussenpeiling toegevoegd.
2.8	03-04-2023	R. Loves	Aangepast aan nieuwe studiejaar.

2 Toetsing

Om te beoordelen of je de doelstellingen van deze module hebt behaald, krijgt de groep een cijfer voor het geleverde eindproduct. De docent gebruikt hierbij de 'Scoring Rubrics' uit bijlage 7.2.

Naast het eindproduct dient er een presentatie te worden gegeven over het eindproduct. Hier is het vooral belangrijk om het proces en ontwerp toe te lichten. De beoordelingscriteria voor de presentatie zijn te vinden in bijlage 7.3.

Aan de hand van dit cijfer moeten de studenten zelf onderling de individuele cijfers gaan bepalen (want zij weten precies hoe andere leden hebben gefunctioneerd in de groep) op basis van een aantal factoren. Dit cijfer zal als advies dienen voor de docent. Hoe dit te werk gaat is de zien in bijlage 7.4. **Let wel, de docent is te allen tijde verantwoordelijk voor het eindcijfer en hij/zij mag dan ook afwijken van het gegeven studentenadvies.**

Het is bekend dat studenten het lastig vinden om elkaar onderling te beoordelen. Toch is het belangrijk dat je dit kunt, omdat de kans groot is dat je daar ook mee in aanraking komt in de latere beroepspraktijk. De beoordeling mag je dan ook anoniem doen. Vanzelfsprekend worden onderdelen aangedragen waarop je moet letten tijdens de beoordeling.

Tabel 2.1 geeft een overzicht van de eindtoets met relevante informatie over de normering, punten en credits.

Tabel 2.1 Overzicht toetsing

Methode	Max. punten	Norm-%	Norm in punten	Credits	Deadline
Eindopdracht + presentatie		55%	55	3EC	Week 8
Totaal				3EC	

2.1 Actieve participatie

Voor deelname aan en beoordeling van deze module is actieve participatie vereist.

Zie de studiegids van jouw opleiding voor aanvullende regelgeving hieromtrent.

2.2 Module herkansing

Studenten die bij de eerste beoordeling een onvoldoende gescoord hebben, kunnen vragen om een herkansing. Voor de actieve participatie geldt onder bepaalde voorwaarden een vervangende opdracht, de opdracht zelf kan na beoordeling en feedback aangepast worden en voor een 2e beoordeling worden aangeboden.

De herkansing moet binnen 1 week worden aangevraagd bij de docent welke verantwoordelijk is voor het project. De herkansingsopdracht(en) moet in de eerste of tweede week van **module periode 5** worden ingeleverd/gepresenteerd.

3 Programma

Eén van meest voorkomende beroepen in het werkveld van de ICT is software engineer. De basis voor software engineers (maar ook ander ICT'ers) is een gedegen kennis van (objectgeoriënteerde) programmeertalen.

Als C# programmeur ga je (meestal) in teamverband samenwerken in een team van (ervaren) C# programmeurs. Je wordt mede verantwoordelijk voor het onderhouden en door ontwikkelen van bestaande pakketten en helpt bij het bouwen en testen van nieuwe applicaties voor zowel intern en extern gebruik.

Deze module bevat een aantal (wekelijkse) werkcolleges. De student dient in groepsverband zelfstandig door te werken. Dit dient te gebeuren door middel van zelfstudie in combinatie met werkcolleges.

Tijdens de werkcolleges heeft de student de mogelijkheid om problemen met de opdrachten te bespreken met de docent en tevens kunnen deze dan ingeleverd worden.

3.1 Collegevormen

Hieronder worden de verschillende college types omschreven.

3.1.1 Werkcolleges

Het is de bedoeling dat de student door middel van zelfstudie de werkcolleges voorbereid en daar eventuele vragen kan stellen. Ook worden werkcolleges gebruikt om de voortgang van de student te toetsen.

3.2 Programmaoverzicht

In onderstaand overzicht is een overzicht te vinden van de wekelijkse activiteiten.

Week	Taak. Nr.	Studie activiteit
1	3.3.1	Introductiecollege
	3.3.2	Hoorcollege 1 (Uitleg WPF)
	3.3.3	Werkcollege 1 (Ontwikkelen concept voor een applicatie/Uitzoeken materiaal/Planning/Beginpeiling).
2	3.3.4	Werkcollege 2 (Opleveren startdocument)
3	3.3.5	Werkcollege
4	3.3.6	Werkcollege 4 (Tussenpeiling/Opleveren klassendiagram)
5 t/m 8	3.3.5	Werkcollege
9	3.3.7	Presentatie applicatie

3.3 Weekprogramma's

3.3.1 Introductiecollege

Week	1
Werkvorm	Hoorcollege
Duur	1
Lesdoelen	<ul style="list-style-type: none">> De student verkrijgt een overzicht van de inhoud van de module Programmeren in C# 2.> De student doorziet wat de werkwijze en de beoordeling binnen de module Programmeren in C# 2 zijn.
Inhoud	Tijdens het introductiecollege krijg je instructie over onder andere de werkwijze/werkvormen, beoordeling, materiaal en inhoud van de module Programmeren in C# 2.
Vorbereiden	
Individuele opdrachten	

3.3.2 Hoorcollege 1

Week	1
Werkvorm	Hoorcollege
Duur	1
Lesdoelen	<ul style="list-style-type: none">> De student verkrijgt inzicht in de basis van het ontwikkelen van een applicatie.
Inhoud	Tijdens dit hoorcollege wordt aan de hand van een demonstratie de werking en mogelijkheden getoond van een applicatie.
Vorbereiden	Voorstel applicatie. Onderzoeken benodigde theorie.
Individuele opdrachten	

3.3.3 Werkcollege 1

Week	1
Werkvorm	Werkcollege
Duur	2
Lesdoelen	> Vaststellen kennishiaten.
Inhoud	Tijdens dit werkcollege kan de student aan de slag met het uitwerken van het concept voor de applicatie. Er wordt begonnen met het opstellen van een startdocument. Dit startdocument bestaat uit het idee, de uitwerking van een aantal schermen (minimaal 4), de planning en de literatuurlijst.
Voorbereiden	Uitwerken startdocument.
Individuele opdrachten	

3.3.4 Werkcollege 2

Week	2
Werkvorm	Werkcollege
Duur	2
Lesdoelen	<i>(Door student in te vullen)</i>
Inhoud	<i>(Door student in te vullen)</i>
Voorbereiden	Definitief voorstel voor een app, opleveren startdocument.
Individuele opdrachten	<i>(Door student in te vullen)</i> Tijdens dit werkcollege zal ook de beginpeiling per team worden gehouden (Go/No Go).

3.3.5 Werkcollege 3, 5 t/m 8

Week	3, 5 t/m 8
Werkvorm	Werkcollege
Duur	2
Lesdoelen	<i>(Door student in te vullen)</i>
Inhoud	<i>(Door student in te vullen)</i>
Voorbereiden	Uitwerken klassendiagram en applicatie.
Individuele opdrachten	<i>(Door student in te vullen)</i>

3.3.6 Werkcollege 4

Week	4
Werkvorm	Werkcollege
Duur	2
Lesdoelen	<i>(Door student in te vullen)</i>
Inhoud	<i>(Door student in te vullen)</i>
Voorbereiden	Tijdens dit werkcollege zal de tussenpeiling worden gehouden.
Individuele opdrachten	<i>(Door student in te vullen)</i>

3.3.7 Presentatie applicatie

Week	9
Werkvorm	Presentatie
Duur	-
Lesdoelen	<i>(Door student in te vullen)</i>
Inhoud	Presentatie Applicatie: <ul style="list-style-type: none">> (Algemene) uitleg App.> Uitleg vertaling klassendiagram naar programma.> Collega teams testen de ontwikkelde App.> Feedback.
Voorbereiden	<i>(Door student in te vullen)</i>
Individuele opdrachten	<i>(Door student in te vullen)</i>

4 Structuur & Organisatie

Het onderstaande schema geeft een overzicht van alle contacturen in deze module.

Daarnaast wordt van studenten verwacht dat zij hun eigen (project)bijeenkomsten plannen waar zij kunnen werken aan de opdrachten. Dit geldt tevens voor de tijd die de student nodig heeft om (individuele) opdrachten voor te bereiden en te maken. Ook deze uren zijn verwerkt in onderstaand schema, waardoor duidelijk wordt weergegeven wat de verwachte studiebelasting per student is.

Tabel 4.1 Studentcontacturen (SCU) en Studiebelastingsuren (SBU) per week:

Werkvorm	Week 1		Week 2		Week 3		Week 4		Week 5		Week 6		Week 7		Week 8		Week 9		Totaal	
	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU
Hoorcollege	1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3
Werkcollege	2	6	2	6	2	6	2	6	2	6	2	6	2	6	2	6	0	0	16	48
Eindtoets	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	1	3
Zelfstudie	0	2	0	2	0	2	0	2	0	2	0	2	0	3	0	3	0	0	0	18
Totale studieactiviteit	3	11	2	8	2	8	2	8	2	8	2	8	2	9	2	9	1	3	18	72
Extra curriculaire activiteiten																			0	12
Eindtotaal																			18	84

SCU = Studentcontacturen (45 minuten)
SBU = Studiebelastingsuren (60 minuten)

5 Literatuur / software

Hieronder wordt vermeld welke literatuur en programmatuur nodig is om het vak met goed gevolg te kunnen doorlopen en af te ronden.

5.1 Verplichte leerstof

In de module Programmeren in C# 2 bepaalt de student zelf de benodigde lesstof om een applicatie te ontwikkelen. In de beginpeiling zal in overleg met de docent vastgesteld worden of de leerstof voldoende is.

5.2 Programmatuur

- > Microsoft Visual Studio 2019 of hoger.
 - Visual Studio Code wordt afgeraden.
- > Git, SVN of ander versiebeheersysteem.

6 Module evaluatie

De module zal worden geëvalueerd door middel van een vragenlijst aan het eind van de module. Deze vragenlijst bevat alle onderdelen van de module inclusief organisatorische aspecten, inhoud, kwaliteit van onderwijzend personeel, etc.

Wij willen je vriendelijk verzoeken deel te nemen aan deze evaluatie. De resultaten van deze evaluatie worden gebruikt om de volgende versie van deze module te verbeteren.

7 Bijlagen

7.1 Projectopdracht

Het is de bedoeling om met C# een applicatie te ontwikkelen. Applicatie is in dit geval een erg breed concept. Vaak wordt er voor deze course onderscheid gemaakt in twee categorieën:

- > Applicaties met een standaard Windows GUI;
- > Spellen.

Voor deze opdracht geldt dat beide goede applicaties zijn. Daarnaast is in overleg met de docent heel veel mogelijk. Als je dus een idee hebt, overleg het dan zeker met de docent.

De eerste week van de module kan worden gebruikt om uit te zoeken welke kennis nodig is om een applicatie te realiseren. In de tweede week zal de beginpeiling plaatsvinden. Daarvoor dient een beknopt plan (**startdocument**), inclusief planning, te worden opgesteld, zodat het door de studentgroep gekozen project beoordeeld kan worden op haalbaarheid/moeilijkheid. Denk hierbij ook aan de keuze voor bepaalde frameworks, deze beïnvloeden namelijk de haalbaarheid/moeilijkheid.

Tussenpeiling

In de 4e week van de module zal er een tussenpeiling plaatsvinden om te beoordelen hoe de voortgang van de App vordert. **Dit wordt gedaan d.m.v. een presentatie van 5 minuten in het responsiecollege.** Hierbij dient een volledig **klassendiagram** van het softwareontwerp gepresenteerd en ingeleverd te worden.

Punten die in de tussenpeiling aan bod moeten komen zijn:

- Waar kwam je vandaan;
- Waar ben je nu;
- Waar ga je naar toe;
- Klassendiagram.

Eindpresentatie

Aan het eind van de module C# 2 **moet** er een eindpresentatie met een demo van de App worden getoond. In de presentatie moet ook de vertaling van het klassendiagram naar programmacode aan bod komen.

Aanvullende algemene eisen

- > Er moet door de studentgroep een startdocument worden opgeleverd, waarin de groep, het idee, welke schermen ontwikkeld gaan worden (minimaal 4) en de literatuur die gebruikt gaat worden uiteenzet. Tevens dient er een planning opgesteld te worden voor de module C# 2.
- > Voor de te ontwikkelen App moet een UML klassendiagram worden ontwikkeld. Het klassendiagram moet door de begeleidend docent worden goedgekeurd voordat er mag worden gestart met het programmeren van de App.
- > Versiebeheer van code moet gedaan worden d.m.v. een daarvoor bestemde, moderne, tool. Denk hierbij aan Git of Subversion. Tools als Google Drive of OneDrive zijn geen goede versiebeheertools voor code.
- > De App moet worden ontwikkeld met Visual Studio 2019 of hoger.

Aanvullende eisen voor spellen

- > Bij de keuze voor een engine/framework moet er in C# .NET geprogrammeerd worden.
 - VB: Unity is niet toegestaan. Het gebruikt C# als taal, maar niet het .NET framework.
- > Het gebruik van geautomatiseerde tools voor het maken van levels, assets en andere benodigheden is toegestaan, maar zal niet worden gezien als werk dat een groep heeft verricht. Hierdoor wordt het niet meegenomen in de beoordeling.

Aanvullende eisen voor applicaties met een standaard Windows GUI

- > Voor het maken van een GUI dient gebruik gemaakt te worden van WPF, UWP, WinUI2 of WinUI3. MAUI mag ook, maar let op: dit is een experimenteel UI framework!
- > Bij het maken van een UI moet, zover mogelijk, gebruik gemaakt worden van de concepten "databinding" en "MVVM". Dit betekent dat de UI alleen code-behind heeft met betrekking tot UI zaken.

7.2 Scoring Rubrics

Onderdeel	Beschrijving	Max. aantal punten
Gebruikersinteractie: Analyseren	Onderbouwing Wireframes of UI schetsen.	5
Gebruikersinteractie: Ontwerpen	Een gebruikersinterface ontwerpen aan de hand van zelfgemaakte schetsen. (Wireframes of UI schetsen).	5
Gebruikersinteractie: Realiseren	Basisprincipes van design toepassen bij het bouwen van eendimensionale communicatiemiddelen. (UI realiseren).	10
Software: Analyseren	MoSCoW analyse and requirements analyse.	10
Software: Ontwerpen	Met gebruik van een ontwerptool een volledig ontwerp opstellen voor een informatiesysteem dat bestaat uit meerdere subsystemen en aansluit op bestaande software. (Class diagram + UI structuur).	15
Software: Realiseren	Een informatiesysteem, dat bestaat uit meerdere subsystemen, bouwen, testen en beschikbaar stellen. Daarbij aansluiten op bestaande software en gebruikmaken van een ontwikkelomgeving.	30
Software: Beheren	Beheren van persoonlijke bestanden en de configuratie van deze bestanden in een software ontwikkelomgeving. Hierbij hoort ook documentatie en code conventies.	15
Beroepsproduct: Presentatie	Cijfer van de gehouden presentatie.	10
Total		100

Het eindcijfer is te berekenen door het behaalde aantal punten te delen door 10.

7.3 Scoring Rubrics Presentatie

A Introductie van de presentatie							
Pakkende opening	Niet						Wel
Voorstellen	Niet						Wel
Introductie onderwerp	Niet						Wel
Overzicht (Opbouw presentatie, duur , omgang vragen)	Niet						Wel
Opmerkingen:							Deelcijfer A:
B Inhoud van de presentatie							
Goed te volgen	Niet						Wel
Goede balans tussen hoofdlijnen en details	Niet						Wel
Goede overgangen tussen de onderdelen	Niet						Wel
Houdt rekening met het niveau de toehoorders	Niet						Wel
Vat waar nodig samen	Niet						Wel
Visuele ondersteuning	Niet						Wel
Opmerkingen:							Deelcijfer B:
C Beëindiging presentatie							
Samenvatting (inclusief kerngedachte) en/of pakkende afsluiting	Niet						Wel
Bedanken toehoorders	Niet						Wel
Opmerkingen:							Deelcijfer C:
D Persoonlijke performance							
Houding	Niet						Wel
Tempo	Niet						Wel
Variatie stemgebruik	Niet						Wel
Kijkt naar het publiek / oogcontact	Niet						Wel
Interactie met het publiek	Niet						Wel
Staat boven de stof	Niet						Wel
Houdt zich aan de tijd	Slecht						Goed
Opmerkingen:							Deelcijfer D:
E Demonstratie							
De keuze voor een demonstratie is passend	Slecht						Goed
Goed voorbereid	Slecht						Goed
Goed uitgevoerd	Slecht						Goed
Opmerkingen:							Deelcijfer E:

F Beantwoorden van de vragen en argumenteren						
De vragen worden goed begrepen, in de context geplaatst en goed geïnterpreteerd	Niet					Wel
De antwoorden zijn helder, gaan concreet en doelgericht in op de vragen	Niet					Wel
De antwoorden geven blij van inzicht in de theoretische achtergronden	Niet					Wel
Opmerkingen:						Deelcijfer F:

Algemene opmerkingen:

Met Demo		
A	*2	
B	*2	
C	*1	
D	*2	
E	*2	
F	*3	
Totaal:		
/12		
Eindcijfer		

7.4 Scoring Rubrics Eindcijfer

Om het individuele eindcijfer te berekenen moeten de studenten van de groep elkaar beoordelen. Dit gaat als volgt:

De docent heeft bepaald dat het gemiddelde geleverde werk van de groep een 7 is.

Als er 4 groepsleden zijn dan krijg je een totaal van $4 \times 7 = 28$ punten. Deze 28 punten mag je per groepslid (minimaal 1, maximaal 10 en met een verschil van maximaal 2 van het docenten) verdelen over alle groepsleden (dus inclusief jezelf). Je kunt nu bijvoorbeeld als cijfers geven: 6, 6.5, 7.5 en 8.

Vervolgens worden van alle groepsleden de gegeven cijfers opgeteld en worden deze gemiddeld. Wijkt een door een student gegeven cijfer meer af dan +/- 1,5 punt van het gemiddelde dan wordt de gehele beoordeling van deze student niet meegenomen en wordt het gemiddelde opnieuw berekend. Dit gebeurt net zo lang tot de afwijking binnen 1,5 punt ligt van het gemiddelde. (Mocht een student bewust zichzelf een te hoog cijfer geven of een andere student een te laag cijfer dan heeft dit geen invloed. Bijkomend verschijnsel is dat de bewuste student geen invloed meer heeft op het cijfer. **Let op dat je voor de gegeven cijfers een onderbouwing geeft. Zonder onderbouwing geldt dat de gegeven beoordelingen niet worden meegenomen.**

Voorbeeld:

Groepscijfer: 7,3

Leden: 7

Totaal te verdelen punten is $7,3 \times 7 = 51,5$.

Verdeel dit totaal aantal punten persoon over alle groepsleden inclusief jezelf.

Naam student	Cijfer
Pieter, Hans	
Hansema, Pieter	
Stijker, Albert	
Hoiting, Barbera	
Leffers, Eric	
Weerts, Bram	
Wighers, Reinier	
Totaal	51,1

